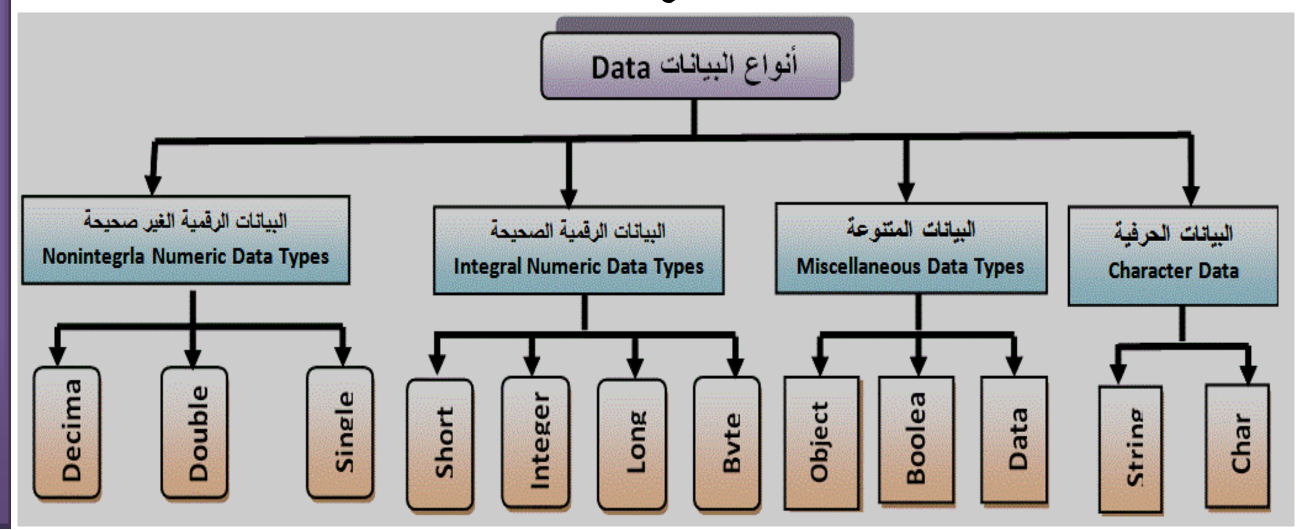


الفصل الاول البيـــــــــانات DATA

أنــــواع البيانات :-

تتعامل لغة Visual Basic مع العديد من انواع البيانات والتى يقوم المستخدم بإدخالها وتخزينها مؤقتاً فى ذاكرة الكمبيوتر وتكون هذه البيانات ناتجة من تنفيذ الأوامر والتعليمات الخاصة بالبرنامج .



١- البيانات الحرفية Character Data Types:

- Char: تضم حرف واحد فقط وكل حرف يخزن بـ ٢ بايت.
- String: يضم جميع الأرقام والحروف التى لا تدخل فى العمليات الحسابية مثل الرقم القومى ورقم التليفون ويشغل ٤ بايت فى الذاكرة.

٢ـ البيانات الرقمية Numeric Data types : وتنقسم الى

ا) بيانات رقمية صحيحة Integral numeric types :

- Byte: من الصفر الى ٢٥٥ تحتاج واحد بايت.
- Short : رقم قصير يضم ٥ أرقام ويحتاج ٢ بايت.
- Integer: رقم صحيح يضم ١٠ أرقام يحتاج ٤ بايت وأسرع من long فى العمليات الحسابية.
- Long: رقم طويل تضم اكثر من ١٩ رقم وتحتاج ٨ بايت بالذاكرة.

ب) البيانات الرقمية الغير صحيحة No integral numeric type :

- Single: رقم فردى تحتاج ٤ بايت.
- Double: رقم زوجى تحتاج ٨ بايت.
- Decimal: يحتاج ١٦ بايت وسريع فى الحسابات المالية.

٣ـ البيانات المتنوعة Miscellaneous Data types :

- Data: يستخدم نوع البيان فى التاريخ فقط ويحتاج ٨ بايت.
- Boolean: يستخدم نوع البيان فى البيانات التى قيمها true اوFalse وتحتاج ٢ بايت.
- Object: تستخدم فى جميع أنواع القيم رقمى او حرفى يحتاج ٤ بايت.

- كل نوع بيان له طريقة تخزن فى ذاكرة الكمبيوتر Ram .
- يرجع أختلاف أنواع البيانات فى الاساس الى الاستخدام الأمثل للذاكرة من حيث المساحة التخزينية والعمليات الحسابية والمنطقية على القيم .
- كل نوع بيان له حيز تخزين فى ذاكرة الكمبيوتر مثل integer تحتاج ٤ بايت .
- كل نوع بيان يحتاج مدى معين مثل Byte تبدأ من الصفر الى ٢٥٥ ولو رقم سالب أو رقم به كسر ياخذ الرقم الصحيح فقط مثل ٥.٢٥ ياخذ ٥ فقط .
- الارقام الغير صحيحة تقبل الصحيح والغير صحيح والارقام الصحيحة تقبل الصحيح فقط .



عبارة عن أماكن محجوزة بذاكرة الكمبيوتر (Rem) عند الاعلان عنها يتحدد لها نوع بيان (Data Type) تأخذ قيمة ثابته لا تتغير أثناء سير البرنامج مثل بعض الثوابت الرياضية كقيمة ط او ثوابت فى الفيزياء كعجلة الجاذبية الأرضية وسرعة الضوء وسرعة الصوت فهى قيم ثابتة لا تتغير .

شروط تسمية الثوابت و المتغيرات :

١ - أن يبدأ اسم الثابت أو المتغير بحرف أو علامة الشرطة السفلى (-) .
٢- ألا يحتوى الاسم على الرموز والعلامات الخاصة مثل (؟ ، * ، ^ ، - ، + ، .) وغيرها.
٣- يمكن أن يتكون من حروف وارقام وعلامة (-) الشرطة السفلى .
٤- لا تستخدم الكلمات المحجوزة للغة الفيجول بيزيك مثل (Dim – Single – As - Douple- me –End) .
٥- يفضل أن يعبر اسمه عن محتواه .
٦- يجب اختيار نوع بيان مناسب لطبيعة البيانات

الإعلان عن الثـــوابت :-

Const	أسم الثابت	AS	Data type نوع البيان	=	القيمة

Const : أمر الاعلان عن الثوابت
Constant_Name : اسم الثابت.
Data Type : نوع البيان المخُزن في الثابت.
Value : القيمة الثابتة التي سيتم تخزينها في الثابت المٌعلن عنه.

مثال ١: تم اعلان عن ثابت C-Name له نوع بيان String وتم تخصيص القيمة النصية أثناء الاعلان "برمجيات آمال مصطفى "

Const	C-Name	AS	String	=	" برمجيات آمال مصطفى "

مثال ٢ : تم الاعلان عن ثابت باسم (Pi) له نوع بيان رقمى عشرى (Single) وتم تخصيص القيمة الرقمية 22/7 او ٣.١٤ أثناء الأعلان .

Const	Pi	AS	Single	=	22/7
Const	Pi	AS	Single	=	3.14

مثال ٣ : تم الاعلان عن ثابت Birth Date له نوع بيان (Date) وتم تخصيص القيمة التاريخ # 1/25/2011 # أثناء الاعلان .

Const	Birth Date	AS	Date	=	# 1 / 25 / 2011 #

لاحـــظ أنـــ :

- تستخدم علامة " " عند كتابة نص .
- تستخدم علامة # # عند كتابة تاريخ أو وقت .

الاعلان عن الثـــوابت:

تصميم برنامج مساحة ومحيط الدائرة وضع أداة ١ text Box و ٣ Button و ١ label كما بالشكل وتغير خاصية Text كما سبق بالترم الأول .




```
Private Sub Button1_Click(ByVal sender
    Dim Radius As Single
    Const Pi As Single = 22 / 7
    Radius = TextBox1.Text
    Label2.Text = Pi * Radius ^ 2
End Sub

Private Sub Button2_Click(ByVal sender
    Dim Radius As Single
    Const Pi As Single = 22 / 7
    Radius = TextBox1.Text
    Label2.Text = Pi * Radius * 2
End Sub
```

كود Button 1 يتم الاعلان عن المتغير (Radius) ثم الاعلان عن الثابت (Pi) ثم تخصيص قيمة من الخاصية (Text box 1 . text) للمتغير (Radius) ثم تخصيص التعبير (Pi * Radius ^2) للخاصية (label2.text) تطبيقاً للقانون لحساب مساحة الدائرة.

كود Button2 يتم الاعلان عن المتغير والثابت كما سبق وعند حساب محيط الدائرة يتم تخصيص للتعبير (Pi * Radius *2)

لاحــظ أنــ :- يمكن الاعلان عن المتغيرات والثوابت على مستوى التصنيف ولا نحتاج للاعلان مرتين كما بالمثال .

تحديد نطاق إعلان المتغيرات والثوابت

عند الاعلان عن ثابت او متغير على مستوى الاجراء Buttn1 وكتابة كود اخر لمساحة الدائرة ولم نعلن عن متغير او ثابت على مستوى الاجراء Button2 سوف يحدث خطا بالبرنامج لذلك لابد من عند استخدام متغير او ثابت اكثر من مره يتم الاعلان عنهم على مستوى التصنيف

```
Private Sub Button1_Click(ByVal sender As Obj
    Dim Radius As Single
    Const pi As Single = 22 / 7
    Radius = TextBox1.Text
    Label2.Text = pi * Radius ^ 2
End Sub

Priv[Name 'Radius' is not declared.]k(ByVal sender As Obj
    Radius = TextBox1.Text
    Lab[!].Text = pi * Radius * 2
End Sub
```

شكل (١-٦) كود (Code) حساب مساحة الدائرة ومحيطها

فســوف تظهر رســالة خطأ مفادها عدم التعرف على المتغير (Radius) والثابت (pi)، حيث تم الإعلان عنهما في الإجراء الخاص بزر الأمر Button1، وبالتالي فهما غير معرفين في الإجراء الخاص بزر الأمر Button2.

ويمكنك التأكد من ذلك أيضًا من المستطيل الأصفر إذا وجهت مؤشر الفأرة إلى المتغير (pi) في السطر الثاني كما هو موضح بشكل (١-٧):

```
Private Sub Button2_Click(ByVal sender As Obj

    Radius = TextBox1.Text
                    [Name 'pi' is not declared.]
    Label2.Text = pi * Radius * 2
                  [!]
End Sub
```

ولحل هذا المشكلة نقوم بالاعلان عن متغير و ثابت مرتين كما بالمثال السابق لكن نستطيع تجنب ذلك الخطا والتكرار مرتين بكتابة كود الاعلان عن متغير وثابت على مستوى التصنيف كالتالى :

```
Public Class Form3
    Dim Radius As Single
    Const pi As Single = 22 / 7
    Private Sub Button1_Click(ByVal sender As
        Radius = TextBox1.Text
        Label2.Text = pi * Radius ^ 2
    End Sub

    Private Sub Button2_Click(ByVal sender As
        Radius = TextBox1.Text
        Label2.Text = pi * Radius * 2
    End Sub
End Class
```



عبارة عن أماكن محجوزة بذاكرة الكمبيوتر **Ram** عند الاعلان عنها يتحدد لها اسم ونوع **Data Type** وعادى ما تتغير قيتها أثناء سير البرنامج ويمكن أن ياخذ المتغير قيمة ابتدائية تسمى **Initial** ثم تتغير قيمته أثناء سير البرنامج مثل سعر سلعة أو منتج – قيمة ضريبة – عنوان موظف .

إعلان عن المتغيرات

Dim Variable_Name AS Data type نوع البيان = initial value

Dim : أمر الاعلان عن المتغيرات.
Variable Name : اسم المتغير.
Data Type : نوع البيان الذي سيتم تخزينه في المتغير.
Initial Value : القيمة الابتدائية التي يتم تخزينها في المتغير المٌعلن عنه، وهذا الجز اختياري.

مثال ١ : تم اعلان عن متغير باسم **F-Name** له نوع بيان **String**

Dim F-Name AS String

مثال٢ : تم اعلان عن متغير باسم **Total_Price** له نوع بيان **Single**

Dim Total_Price AS Single

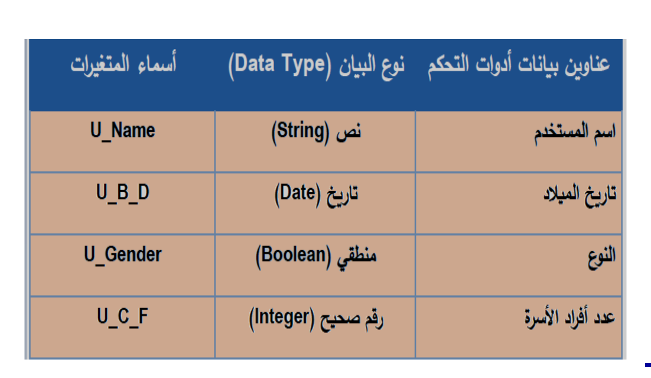
مثال٣ : تم اعلان عن متغير باسم **Today** له نوع بيان تاريخ **Date**

Dim Today AS Date = #1 / 25 / 2017 #

لاحظ : عند اهمال القيمة الابتدائية فأن امتغير ياخذ القيمة الافتراضية
١- المتغير العددى صحيح او عشرى يأخذ القيمة صفر
٢- المتغير الحرفى يأخذ القيمة سلسلة حرفية فارغة

الاعلان عن المتغيرات : قم بتصميم نافذة نموذج لادخال بيانات من المستخدم وكتابة كود الاعلان عن التغيرات داخل Button1 زر التسجيل كما بالشكل





عناوين بيانات أدوات التحكم	نوع البيان (Data Type)	أسماء المتغيرات
اسم المستخدم	نص (String)	U_Name
تاريخ الميلاد	تاريخ (Date)	U_B_D
النوع	منطقى (Boolean)	U_Gender
عدد أفراد الأسرة	رقم صحيح (Integer)	U_C_F

أمثلة للإعلان عن متغير :

```
Private Sub Button1_Click(ByVal sen
    Dim U_Name As String
    Dim U_B_D As Date
    Dim U_Gender As Boolean
    Dim U_C_F As Integer
End Sub
```



جملة التخصيص Assignment :

عبارة عن جملة من طرفين بينهما علامة (=) ولا يقصد بها التساوى الحسابى والطرف الأيسر يمثل اسم المتغير أو الثابت الذى يخزن فيه القيمة بالطرف الايمن

الطـــــــــــرف الإيمن	علامة التخصيص	الطـــــــــــرف الإيسر
قيمة مجردة أو قيمة من متغير (Variable) أو خاصية (Property) أو قيمة من تعبير حسابى (Expression)	=	متغير (Variable) أو خاصية (Property)

أولاً : القيمة المجردة : يتم الاعلان عن المتغير (number) من النوع (Integer) ثم تخصيص القيمة المجردة (5) للمتغير (Number) مثل:

أمثلة للتخصيص:

Dim	Number	AS	Integer
Number	=	5	

ثانياً : القيمة من متغير : يتم الاعلان عن متغيرين (Number 1) و (Number 2) من النوع (Integer) ثم تخصيص القيمة المجردة (5) للمتغير (Number 1) وتم تخصيص قيمة المتغير (Number 1) للمتغير (Number 2)

Dim	Number 1	AS	Integer
Dim	Number 2	AS	Integer
Number 1	=	8	
Number 2	=	Number1	

ثالثاً : قيمة من خاصية : تم الاعلان عن متغير (Name) من النوع (String) ثم تم تخصيص القيمة من الخاصية (text) لأداة التحكم (Label 1)

Dim	Name	AS	String
Name	=	Label1.text	

رابعاً : قيمة من تعبير حسابى : تم الاعلان عن متغير (Number) من النوع (Single) ثم تخصيص القيمة المجردة (٧) للمتغير (Number) ثم غعادة عملية التخصيص للمتغير (Number) ثم تم غعادة عملية التخصيص للمتغير (Number) من خلال تعبير (Number / 3)

```
Dim       Number 1      AS      Single
Number 1     =          7
Number 1     =       Number 1 / 3
```

خامسا : الاعلان عن متغير وتخصيص قيمة فى خطوة واحدة : مثل

```
Dim    Number1    AS    Single   =   5.6
```

- يمكن التخصيص للمتغيرات Variables أثناء الإعلان عنها أو فى مرحلة لاحقة .
- يمكن أعادة التخصيص أى عدد من المرات ولذلك تسمى متغيرات لانها تتغير أثنا سير البرنامج .
- التخصيص يمكن أن يكون بقيمة مجردة أو بمتغير أو بخاصية أو بتعبير حسابى سبق شرحهم
- علامة = لا يقصد بها التساوى الحسابى وأنما يقصد بها علامة للتخصيص مثل X = X + 1 تمثل (X) بالطرف الأيسر مخزن بالذاكرة و (X) بالطرف الايمن تمثل القيمة واذا أفترضنا أن (X) تساوى (5) بعد عملية التخصيص (1 + 5) أى 6 .

```
Me.Label5.Text = UserName &vbCrLf & UserBirthDate & vbCrLf & _UserGender & vbCrLf &
UserNoFamily
```

شرح الكود السابق :

الكود عبارة عن جملة تخصيص لمجموعة من المتغيرات للخاصية Text لأداة التحكم Label 5

- يفصل بين كل متغير والآخر (& vbcrlf &)
- معامل الربط & : يستخدم للربط بين القيم النصية (السلاسل النصية)
- الكلمة المحجوزة vbcrlf : رمز إضافة مفتاح الادخال وسطر جديد
- تستخدم علامة (_) حتى يمكن كتابة الكود على أكثر من سطر وذلك لتنظيم وتسهيل قرءة الكود
- Me تعبر عن نافذة النموذج Form

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal
    Dim U_Name As String
    Dim U_B_D As Date
    Dim U_Gender As Boolean
    Dim U_C_F As Integer

    U_Name = TextBox1.Text
    U_B_D = TextBox2.Text
    If RadioButton1.Checked = True Then
        U_Gender = True
    End If
    If RadioButton1.Checked = False Then
        U_Gender = False
    End If
    U_C_F = TextBox3.Text

    Me.Label5.Text = U_Name & vbCrLf & U_B_D & vbCrLf & _
    U_Gender & vbCrLf & U_C_F

End Sub
```

الإعلان عن المتغيرات

تخصيص قيم للمتغيرات

استخدام المتغيرات

كتابة الملاحظات : تستخدم كلمة (REM) لتساعد القارئ على فهم وتوضيح الأكواد المكتوبة فى نافذةالكود أو العلامة (،) ما يكتب بعدها لا يعتبر أكواد ويهمل عند ترجمة الأكواد بواسطة المترجم .

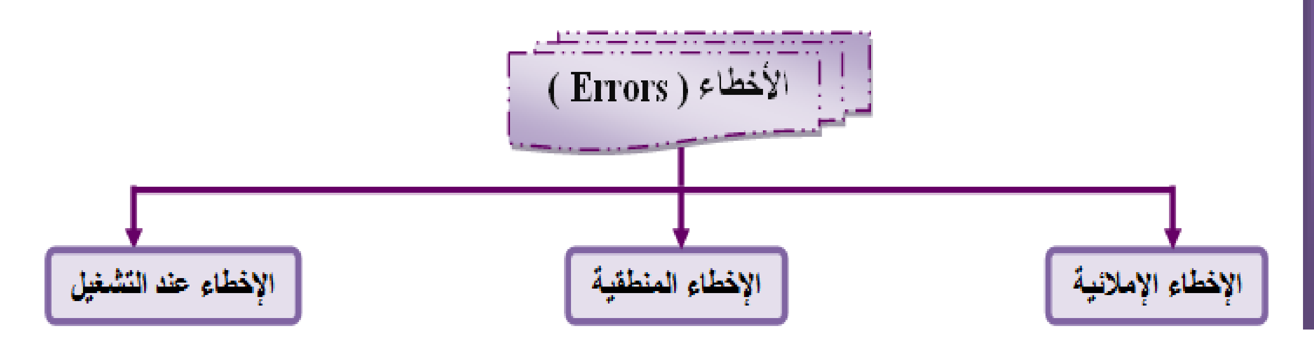
أولويات تنفيذ العمليات الحسابية :

أولويات تنفيذ العمليات الحسابية :

١- فق الأقواس من الداخل الى الخارج
٢- رفع الأسس
٣- الضرب و القسمة من اليسار الى اليمين
٤- الجمع والطرح من اليسار الى اليمين

فى حالة وجود أكثر من عملية حسابية فى نفس الرتبة كالضرب والقسمة أو الجمع والطرح أو الآس مكرر أكثر من مرة فإن ترتيب تنفيذها يكون من اليسار الى اليمين مثل

3^2*2+4*4
9*2+4*4
18+4*4
18+16
=34



أولاً : الإخطاء لغوية Syntax Errors :

هو خطأ يحدث عند كتابة الكود بصورة غير سليمة مثل **Din X As Single** تم الاعلان عن المتغير **X** ولكن تم كتابة **Dim** خطأ . خطأ فى عدم تخصيص قيمة عند الاعلان عن ثابت مثل **const x as single**

معالجة الخطأ : IDE لاتسمح بمثل هذه الاخطا يتم اكتشاف الخطأ عند كتابة الخطأ .

ثانياً : الاخطاء المنطقية Logic Error :

هى أخطأ تودئ الى نتائج غير سليمة عند استخدام تعبيرات بناؤها غير سليم مثل عند كتابة (Pi + Radius ^2) فعند تنفيذ البرنامج لاتظهر أخطا ولكن النتيجة تكون خطأ .

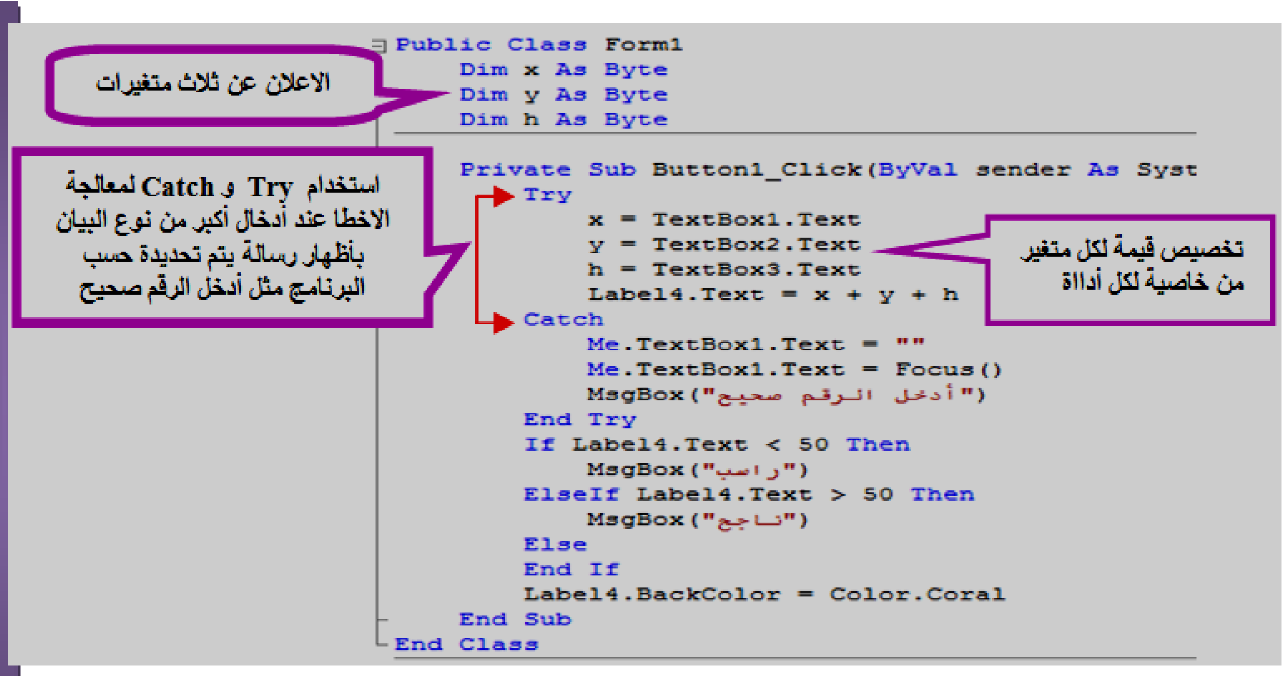
معالجة الخطأ : يجب اختبار البرنامج بأدخال بيانات سبق التأكد من صحتها .

ثالثاً : الأخطاء عند التشغيل Runtime Errors :

هى أخطاء تحدث عند تخصيص قيمة أكبر من مدى نوع البيان المستخدم وتخصيص نوع قيمة يختلف عن نوع بيان المتغير أو الثابت أو الخاصية ولا يستطيع مترجم ترجمتها من الاعلان عن نوع بيان **Byte** وادخال رقم أكبر من ٢٥٥ .

التعامل مع الأخطاء : -

تصميم برنامج لجمع ثلاث درجات لمواد مختلفة والإعلان عن ثلاث متغيرات ولهم نوع بيان **Byte** اى لابد من أدخال من الصفر الى ٢٥٥



```
Public Class Form1
    Dim x As Byte
    Dim y As Byte
    Dim h As Byte

    Private Sub Button1_Click(ByVal sender As Syst
        Try
            x = TextBox1.Text
            y = TextBox2.Text
            h = TextBox3.Text
            Label4.Text = x + y + h
        Catch
            Me.TextBox1.Text = ""
            Me.TextBox1.Text = Focus()
            MsgBox("أدخل الرقم صحيح")
        End Try
        If Label4.Text < 50 Then
            MsgBox("راسب")
        ElseIf Label4.Text > 50 Then
            MsgBox("ناجح")
        Else
        End If
        Label4.BackColor = Color.Coral
    End Sub
End Class
```






معالجة الأخطاء المحتمل وقوعها من خلال جملة :

➡ **Try**
كتابة الكود المحتمل حدوث أخطاء عند تشغيل البرنامج

➡ **Catch**
كتابة رسالة عند عدم تحقق الشرط

الفصـــل الثانى
التفـــــــرع Branching

التفـــرع:- هو اتخاذ القرار أو اختيار تنفيذ مجموعة من الخطوات بناءاً على سؤال معين

التفرع برمجيا

- IF Then
- IF Then Else
- Select Case

الصيغة للتفرع باستخدام If Then:
```
IF التعبير الشرطى Then
الكود فى حالة تحقق الشرط
End if
```

الصيغة للتفرع باستخدام If Then Else:
```
IF التعبير الشرطى Then
الكود الأول
Else وإلا
الكود الثانى
End if
```

الصيغة للتفرع باستخدام Select Case
```
Select Case Variable
Case Value          الحالة الاولى
الامر الذى ينفذ أذا كانت قيمة المتغير مساوية للقيمة ١
Case Value 2          الحالة الاولى
الامر الذى ينفذ أذا كانت قيمة المتغير مساوية للقيمة ٢
Case Value ٣          الحالة الاولى
الامر الذى ينفذ أذا كانت قيمة المتغير مساوية للقيمة ٣
أضف الحالات المطلوبة حسب البرنامج
Case Else
الكود فى حالة عدم تحقق اى شرط من الشروط السابقة
End Select
```

أولاً: للتفرع باستخدام جملة If Then:-

الصيغة للتفرع باستخدام If Then:
```
IF التعبير الشرطى Then
الأوامر التى يتم تنفيذها فى حالة تحقق الشر Code
End if
```

جملة If Then البسيطة تستخدم فى حالة بديل واحد فقط :

١- جملة (If Then) تتكون من If ثم التعبير الشرطى ثم كلمة Then على سطر واحد .
٢- كتابة مجموعة من الأوامر المطلوب تنفيذها إذا كان ناتج الشرط صواب تحقق true .
٣- تنتهى جميع جمل if الشرطية ب End if اى انتهاء جملة الشرط .

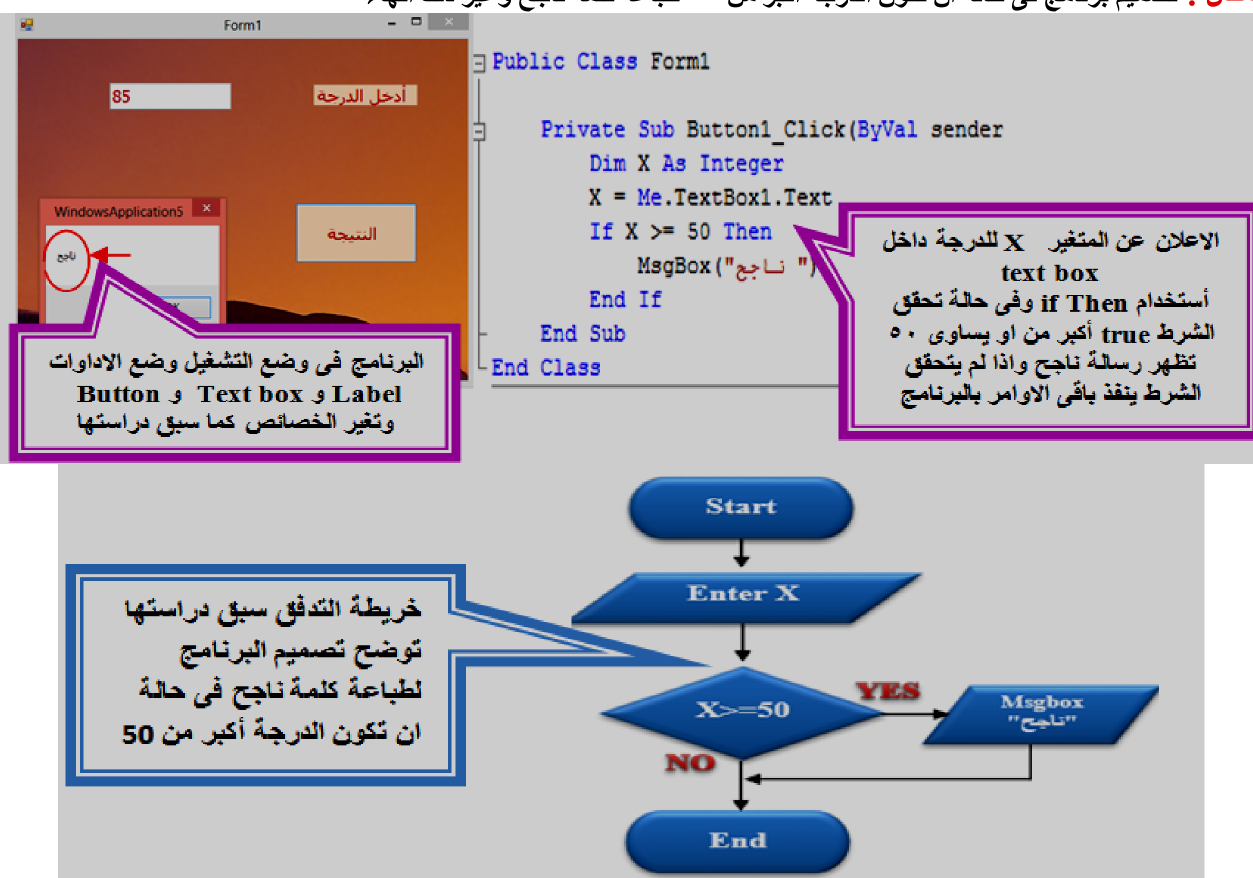
- أذا كانت قيمة الشرط صحيح يتم تنفيذ الأوامر التى تلى Then وحتى End if .
- أذا كانت قيمة الشرط لم يتحقق false يتم تنفيذ الأوامر التى تلى End if اى باقى أوامر البرنامج .

التعبير الشرطى : هو جزء من كود البرمجة يتكون من ثلاثة أجزاء منطقية يسبقها متغير أو ثابت يتم مقارنته مع قيمة مجردة أو قيمة من متغير أو ثابت اخر او ناتج تعبير حسابى فإذا تحقق الشرط فهذا يعنى أن نتيجة التعبير الشرطى True ويتم تنفيذ كود معين حقق الشرط فهذا يعنى ان نتيجة التعبير الشرطى False ويتم تنفيذ كود اخر

١	٢	٣		
متغير أو ثابت مثل A	عامل المقارنة مثل >، <، <>،=	قيمة		
		قيمة متغير أو ثابت	مجردة	ناتج تعبير حسابى

مثال للتعبير الشرطي	التعبير الشرطي Conditional Expression		
	بعد العلاقة المنطقية	يوجد (٦) علامات منطقية	قبل العلاقة المنطقية
If A > 5 If A < 5	قيمة مجردة	> أكبر من Or < أصغر من Or	Variable Or Constant
If B <= A If B >= A	متغير	<= أصغر من أو يساوي Or >= أكبر من أو يساوي Or	
If B = A + 3 * 2 If C <> A – 3 * 2	قيمة من تعبير Expression	= يساوي Or <> لا يساوي	

مثال : تصميم برنامج فى حالة أن تكون الدرجة أكبر من ٥٠ طباعة كلمة ناجح وغير ذلك أنهاء



ملاحظة : عدم ظهور رسالة عند ادخال قيمة أقل من ٥٠ لان ناتج الشرط False فينفذ ما بعد End if وهو End Sub اى انهاء الإجراء

يمكن كتابة نفس جملة IF السابقة فى سطر واحد ولا يتم وضع End if مثل

If x>= 50 then msgBox (" ناجح")

معاملات المقارنة فى الفيجوال بيزيك :-

المعـــامل	معنى المعـــامل
=	يساوى
<>	لا يساوى
>	أكبر من
<	أصغر من
>=	أكبر من أو يساوى
<=	أصغر من أو يساوى

ثانياً: للتفرع باستخدام جملة If Then Else:-

تستخدم فى حالة وجود اختيارين (بديلين)

- أذا كان ناتج التعبير الشرطى "True" يتم تنفيذ الأوامر بالكود الأول .
- أذاكان ناتج التعبير الشرطى :False" يتم تنفيذ الأوامر بالكود الثانى .

الصيغة للتفرع باستخدام If Then Else:

IF التعبير الشرطى Then
الأوامر التى يتم تنفيذها فى حالة تحقق التعبيرالشرطى "الكود الاول "
Else
الأوامر التى يتم تنفيذها فى حالة عدم تحقق التعبير الشرطى "الكود الثانى"
End if

جملة If Then تستخدم فى حالة بديلين :

١- جملة (If Then) تتكون من If ثم التعبير الشرطى ثم كلمة Then على سطر واحد .
٢- ثم الأوامر المطلوب تنفيذها اذا كان ناتج التعبير الشرطى " True " .
٣- ثم كلمة Else على سطر واحد ومعناها " والا " .
٤- ثم مجموعة من الأوامر فى حالة عدم تحقق الشرط الاول كان خطأ "False" .
٥- تنتهى جملة IFالشرطية ب "End if "

- أذا كان ناتج التعبير الشرطى صواب "True" يتم تنفيذ الأوامر التى تلى كلمة "Then" حتى كلمة "Else "
- أذا كان ناتج التعبير الشرطى "false" يتم تنفيذ الأوامر التى تلى "Else" حتى End if

مثال : تصميم برنامج فى حالة أن تكون الدرجة أكبر من ٥٠ طباعة كلمة ناجح وطباعة راسب فى حالة ان تكون الدرجة أقل من ٥٠

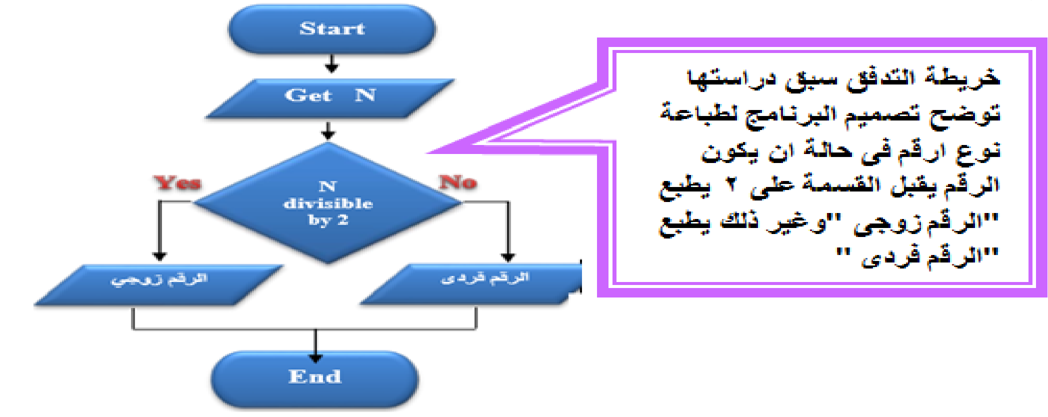


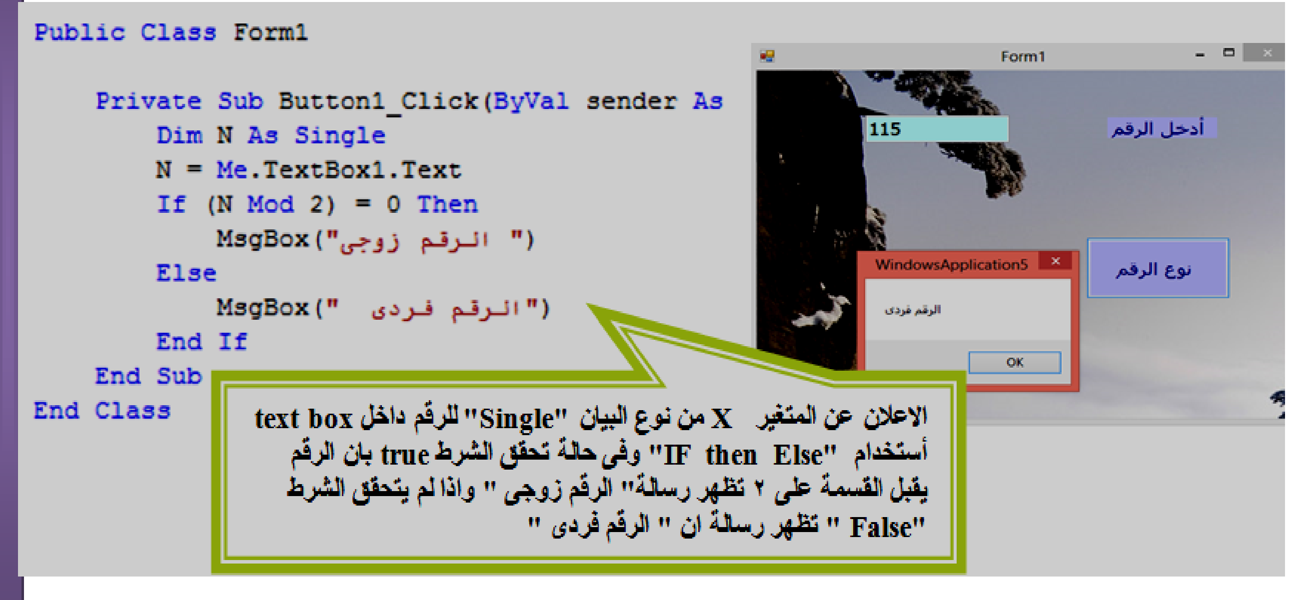


يمكن كتابة جملة IF Then Else على سطر واحد دون كتابة End if

IF X >=50 then msg Box(" ناجح" Else msg Box (" راسب ")

مثال ٢ : تصميم برنامج فى حالة أن تكون الدرجة أكبر من ٥٠ طباعة كلمة ناجح وطباعة راسب فى حالة ان تكون الدرجة أقل من ٥٠ .





رابعاً: للتفرع باستخدام جملة Select Case :-

تستخدم فى حالة وجود بدائل كثيرة جملة Select Case تشبه جملة If then Else If ولكن
١.يفضل استخدامها عندما يكون التفرع لأكثر من شرطشين
٢.يفضل استخدامها عندما يكون التفرع أتخاذ القرار معتمداً على قيمة متغير واحد
٣.تيسر فهم البرنامج بنافذة الكود

الصيغة للتفرع باستخدام Case Select

Select Case Variable
Case Value الحالة الاولى
الامر الذى ينفذ أذا كانت قيمة المتغير مساوية للقيمة ١
Case Value 2 الحالة الاولى
الامر الذى ينفذ أذا كانت قيمة المتغير مساوية للقيمة ٢
Case Value ٣ الحالة الاولى
الامر الذى ينفذ أذا كانت قيمة المتغير مساوية للقيمة ٣
أضف الحالات المطلوبة حسب البرنامج
Case Else
الكود فى حالة عدم تحقق اى شرط من الشروط السابقة
End Select

جملة select Case تستخدم فى حالة وجود أكثر من بديل :

١- جملة (select case) تتكون من Select Case ثم يأتى بعدها المتغير "Variable" .
٢- السطر التالى لـ select case به كلمة case وهى تدل على المتغير ويليها الشرط .
٣- السطر التالى لكلمة case به الأوامر المطلوبة تنفيذها إذا تحقق الشرط .
٤- يمكن تكرار سطر case والأوامر التابعة له لـه اى عدد من الاحتمالات حسب البرنامج .
٥- ثم الشرط الافتراضى وهو سطر Case Else يتم تنفيذ الأوامر التالية لهذا السطر أذا لم يتحقق اى شرط من الشروط السابقة .
٦- جملة (select case) تنتهى ب End select .

```
Private Sub Button1_Click(ByVal sender As System.Object, ByV
    Dim D As Single
    D = Me.TextBox1.Text
    Select Case D
        Case 85 To 100
            Label2.Text = " ممتاز"
        Case 75 To 84
            Label2.Text = (" جيد جدا")
        Case 65 To 74
            Label2.Text = ("  جيد")
        Case 50 To 64
            Label2.Text = (" مقبول  ")
        Case 0 To 49
            Label2.Text = (" ضعيف ")
        Case Else
            Label2.Text = "الرجاء ادخال درجة بين 0 الى 100"
    End Select
```

Form1
95
أدخال الدرجة
ممتاز
النتيجة

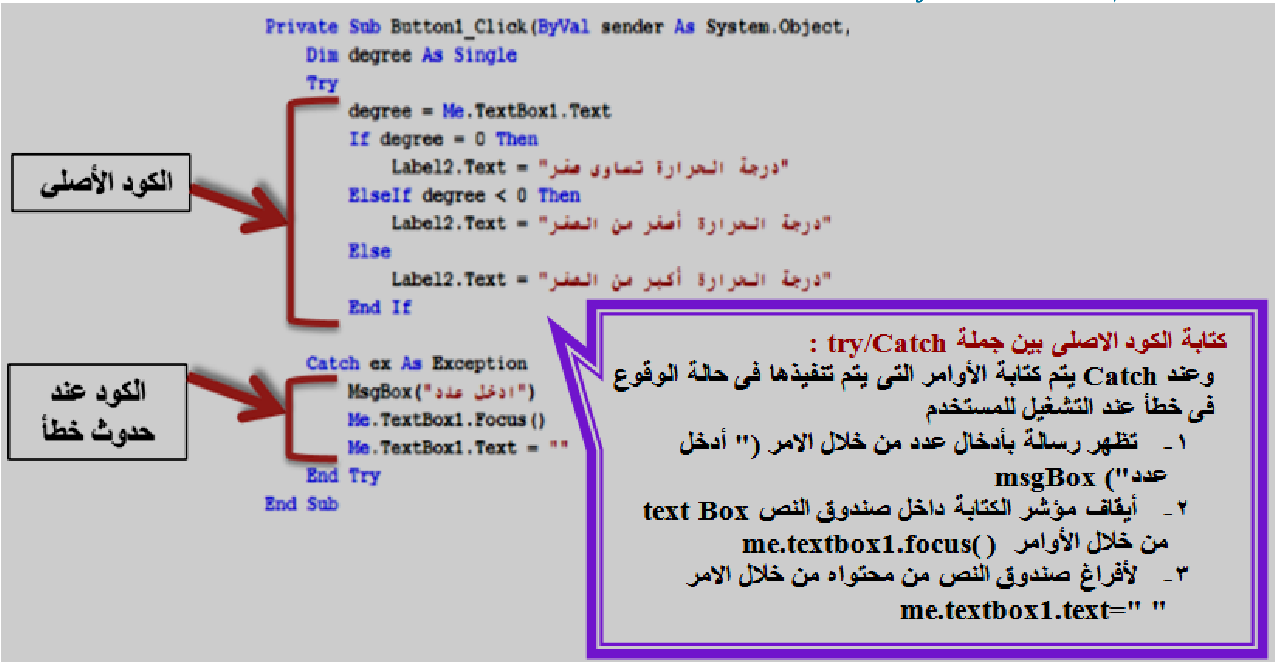
مثال لاستخدام جملة Select case:
عند تحقق التعبير الشرطى يتم عرض التقدير للدرجة داخل text box
واذا لم يتحقق الشرط الاول ينتقل الى الشرط الثانى وعند عدم تحقق الشرط الثاني ينتقل الى الشرط الثالث وهكذا
وعند Case Else يفذ الأوامر فى حالة عدم تحقق اى شرط من الشروط السابقة

الأخطأ المتوقع أن تحدث فى تصميم البرامج السابقة :

1) كتابة درجة باللغة العربية مثل أدخال خمسة عشر بدلآصمن ١٥ .
2) عدم أدخال نص سلسلة حرفية فارغة .
3) أعلان عن نوع بيان Byte وادخال الدرجة أكبر من نوع البيان .

طرق منع وقوع خطأ التشغيل البرنامج :

❖ استخدام جملة Try / catch





أولاً: جملة Next For -: تستخدم فى حالة تكرار كود محدد عدد من المرات :

الصيغة العامة لجملة For Next

```
For variable = start value   TO End value Step Add value
Code
Next
```

١- Variable : اسم المتغير الذى يمثل العداد ويجب ان يكون نوعه رقمى (صحيح أو عشرى) .
٢- Start value : قيمة بداية العداد أو بداية التكرار وهى قيمة رقمية .
٣- End value : قيمة نهاية العداد أو التكرار وهى قيمة رقمية
٤- Add value : قيمة زيادة العداد أو القيمة التى يزيد بها العداد حتى يصل الى قيمة النهاية
٥- Code : عبارة عن أمر أو اكثر المراد تكراره ويكون بين بداية الحلقة التكرارية for ونهايتها Next

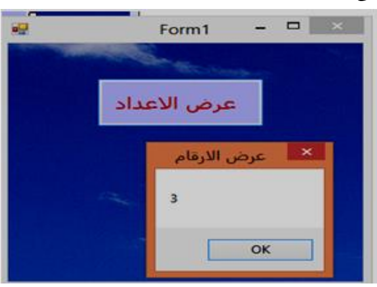
الصيغة العامة لجملة For Next عندما تكون قيمة الزيادة موجبه بواحد

```
For variable = start value   TO End value
Code
Next
```

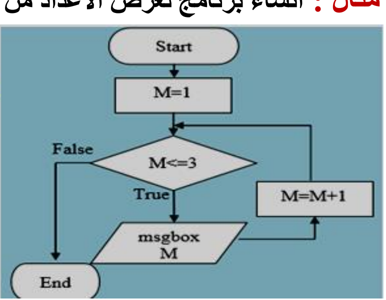
ملاحظة هامة : كتابة اسم المتغير بجوار Next اختيارية

- يمكن جعل قيمة البداية أكبر من قيمة النهاية وتكون قيمة الزيادة متغير العداد بالسالب
- يمكن جعل قيمة البداية أو قيمة النهاية أرقام صحيحة أو عشرية أو متغيرات
- عند الاعلان عن قي عشرية يجب الاعلان عن متغيرات من النوع Single
- قيمة الزيادة الافتراضية فى جملة التكرار Next For فى حالة عدم كتابة Step تكون واحد صحيح

مثال : أنشاء برنامج لعرض الاعداد من ١ الى ٣ عند الضغط على زر عرض الاعداد من ١ الى ٣ يعرض رسالة بالاعداد :




```
Private Sub Button1_Click(ByVal
    Dim X As Integer
    For X = 1 To 3
        MsgBox(X)
    Next
End Sub
```



-١

خطوات سير البرنامج :

١- الاعلان عن المتغير X بنوع بيان رقمى صحيح Dim X As integer
٢- أستخدام كلمة For بداية الحلقة اتكرارية وتخزن القيمة داخل " X " كعداد وتبدأ الحلقة التكرارية من ١ وتنتهى عند ٣
٣- ظهور رسالة بالعدد ١ الذى يمثل X
٤- تقوم Next بمقارنة قيمة المتغير X بقيمة النهاية وعند عدم تحقق الشرط false يتم زيادة X+1
٥- تكرار الرسالة بالعدد ٢ ثم يذهب عند Next ويقارن بالقيمة النهاية وعند عدم تحقق الشرط يقوم بزيادة رقم X+1
٦- ويتم التكرار حتى تصل قيمة X اكبر من قيمة النهاية ٣ ويتحقق الشرط فيخرج من الحلقة التكرارية

مثال ٢ : تصميم برنامج لعرض الاعداد من ١ الى ٣ فى صندوق النص Textbox عند الضغط على زر عرض الاعداد من ١ الى ٣ يعرض الاعداد داخل صندوق النص textbox :




```
Private Sub Button1_Click(ByVal sender As System.O
    Dim m As Integer
    For m = 1 To 3
        Me.TextBox1.Text = Me.TextBox1.Text & m
    Next m
End Sub
```



١- الأمر داخل الحلقة التكرارية والذى يعنى وضـــع قيمة المتغير (M) بجانب ما هو موجود داخل صندوق النص (TextBox)، باستخدام معامل الربط & لربط سلسلتين حرفيتين ببعضهما.

٢- في حالة كتابة السـطر المشـار إليه بهذه الطريقة me.textbox1.text=m لن نلاحظ تغير قيمة المتغير (M) أثناء التنفيذ وإنما سيظهر آخر قيمة فقط وهي ٣ لأن عرض الناتج سيتم في نفس المكان فيظهر آخر قيمة فقط.

كود لمسح محتويات صندوق النص textbox : Me.textbox1.text=""

نلاحظ ظهور الاعداد بجوار بعضهم ١٢٣ ولكى يظهر كل رقم بسطر بالكود التالى :

Me.TextBox1.Text = Me.TextBox1.Text & m & vbCrLf

ضبط قيمة الخاصية Multiline الى القيمة true لتعدد الاسطر داخل textbox

تستخدم vbcrlf لإضافة رمز مفتاح الاخال وسطر جديد وهى اختصار لمصطلح visual Basic Carriage Return Line Feed

كود لمسح محتوى صندوق النص

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As Sy
    Dim m As Integer
    Me.TextBox1.Text = ""
    For m = 1 To 3
        Me.TextBox1.Text = Me.TextBox1.Text & m & vbCrLf
    Next m
End Sub
```



مثال ٣: تصميم برنامج جدول ضرب ٣

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    Dim m As Integer
    Dim product As Integer
    Dim str As String
    Me.TextBox1.Text = ""
    For m = 1 To 12
        str = 3 & "x" & m & "="
        product = 3 * m
        Me.TextBox1.Text = Me.TextBox1.Text & str & product & vbCrLf
    Next m
End Sub
```

Form1

3x1=3
3x2=6
3x3=9
3x4=12
3x5=15
3x6=18
3x7=21
3x8=24
3x9=27
3x10=30
3x11=33
3x12=36

جدول ضرب ٣

١- تم تعريف متغير رقمي صحيح باسم (product) لتخزين حاصل الضرب به مع كل تغيير فى قيمة المتغير (M).

٢- تم تعريف متغير حرفي باسم (str) لتخزين شكل جملة حاصل الضرب بحيث نحصل على الناتج بالشكل التالي (3x1= او 3x2= وهكذا) كسلسلة نصية مع كل تغيير فى قيمة المتغير (M).

٣- تم عرض قيمة المتغير (str) ثم معامل ربط & ثم قيمة المتغير (product) في صندوق النص (TextBox).

٤- يمكن كتابة الكود بدون استخدام هذه المتغيرات (Variables) كما يلي:

شكل حاصل الضرب | حاصل الضرب

```
Dim m As Integer
Me.TextBox1.Text = ""
For m = 1 To 12

    Me.TextBox1.Text = Me.TextBox1.Text & 3 & "x" & m & "=" & 3 * m & vbCrLf
Next m
```

مثال ٤ : تصميم جدول ضرب لاى رقم يتم ادخاله داخل صندوق النص

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As Sy
    Dim m As Integer
    Dim product As Integer
    Dim num As Integer
    Dim str As String
    num = Me.TextBox2.Text
    Me.TextBox1.Text = ""
    For m = 1 To 12
        str = num & "x" & m & "="
        product = num * m
        Me.TextBox1.Text = Me.TextBox1.Text & str & product & vbCrLf
    Next m
End Sub
```

Form1

ادخال رقم جدول الضرب المطلوب

9

9x1=9
9x2=18
9x3=27
9x4=36
9x5=45
9x6=54
9x7=63
9x8=72
9x9=81
9x10=90
9x11=99
9x12=108

عرض جدول الضرب

التحكم فى البداية والنهاية ومقدار الزيادة فى جملة For Next :

أضافة كلمة Step وتعنى مقدار الزيادة ويمكن أن تكون قيمة البداية أكبر من قيمة النهاية على أن تكون قيمة الزيادة سالبة وكما يمكن أن تكون قيمة البداية والنهاية والزيادة بأرقام صحيحة أو عشرية أو متغيرات
الصيغة العامة هى

For Variable = start value to End value Step Add value
For X = 5 to 50 step 2 مثل
الكود
Next (Variable)

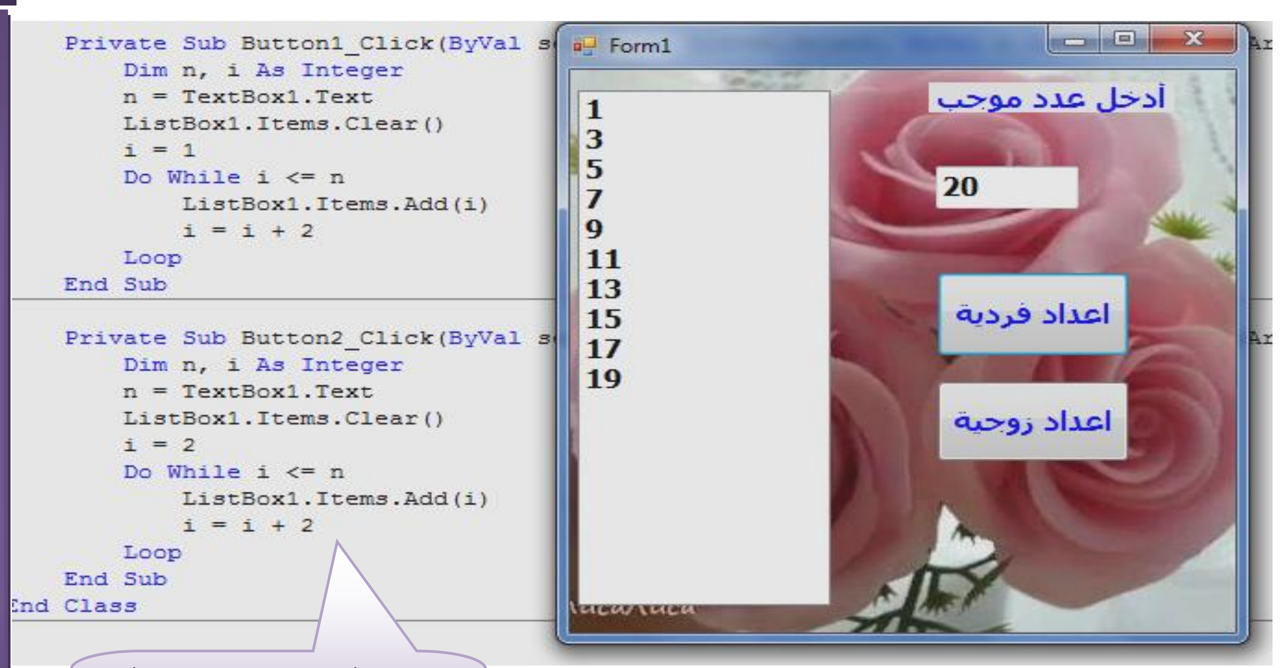
ثانياً: جملة DO While :-

تستخدم فى حالة تكرار كود لعدد من المرات غير معروف نهايته مسبقا او بناءا على شرط :

الصيغة العامة لجملة **Do While**
Do while conditional Expression التعبير الشرطى
Code
Loop

ستم تنفيذ الكود بين بداية الحلقة التكرارية **Do while** ونهايتها **Loop** طالما ان التعبير الشرطى صحيح **True** واذا لم يتحقق الشرط لاى سبب يتم الخروج من الحلقة التكرارية وتنفيذ ما بعد **Loop**

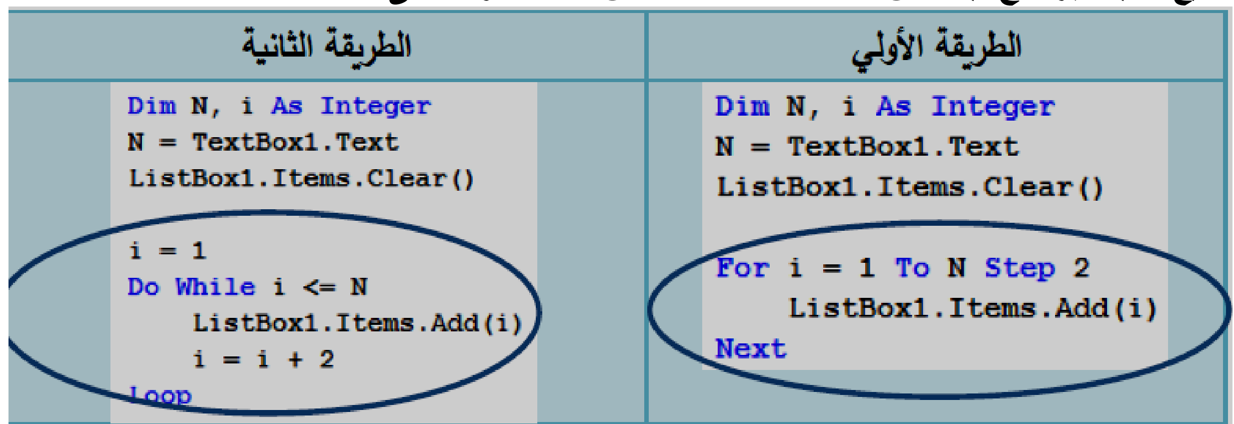
مثال ١ : تصميم برنامج لاظهار الاعداد الفردية مرتبة تصاعديا حتى الرقم الذى يتم ادخاله وعند الضغط على اعداد زوجية يتم عرض الاعداد مرتبة تصاعديا حتى الرقم الذى يتم ادخاله فى صندوق قائمة



الكود الثانة لعرض الاعداد الزوجية
تكون تساوى ٢ إيختلف بقيمة I

لاحظ : لحذف محتويات أداة **ListBox1** من خلال الكود **ListBox1.Items.Clear()**
لاضافة قيمة **I** الى أداة **ListBox1** من خلال الكود **ListBox1.Items.Add(i)**

تستطيع تنفيذ البرنامج ايضا من خلال Next For من خلال الكود التالى



الطريقة الثانية	الطريقة الأولي
Dim N, i As Integer N = TextBox1.Text ListBox1.Items.Clear() i = 1 Do While i <= N ListBox1.Items.Add(i) i = i + 2 Loop	Dim N, i As Integer N = TextBox1.Text ListBox1.Items.Clear() For i = 1 To N Step 2 ListBox1.Items.Add(i) Next

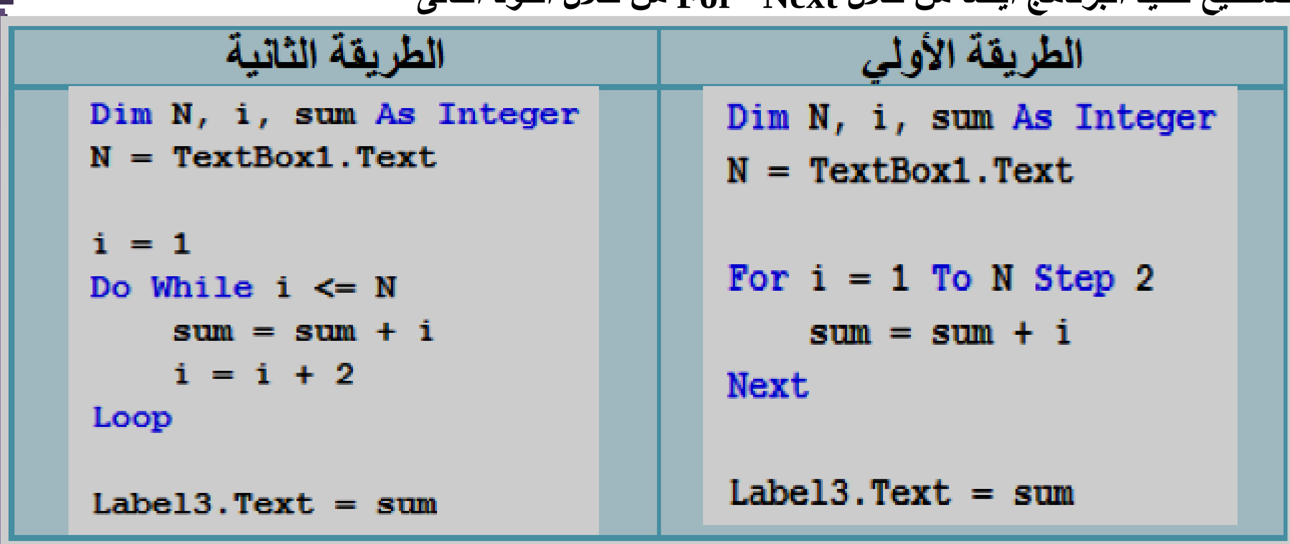
مثال ٢ : تصميم برنامج يستقبل عدد موجب ويعرض مجموع الاعداد الفردية فى Label3 حتى الرقم الذى يتم ادخاله وعند الضغط على مجموع الاعداد الزوجية يعرض الاعداد الزوجية فى Label3 حتى الرقم الذى يتم ادخاله



```
Private Sub Button1_Click(ByVal s
    Dim n, i, sum As Integer
    n = TextBox1.Text
    i = 1
    Do While i <= n
        sum = sum + i
        i = i + 2
    Loop
    Label3.Text = sum
End Sub

Private Sub Button2_Click(ByVal s
    Dim n, i, sum As Integer
    n = TextBox1.Text
    i = 2
    Do While i <= n
        sum = sum + i
        i = i + 2
    Loop
    Label3.Text = sum
End Sub
```


تستطيع تنفيذ البرنامج ايضا من خلال Next For من خلال الكود التالى



الطريقة الثانية	الطريقة الأولي
Dim N, i, sum As Integer N = TextBox1.Text i = 1 Do While i <= N sum = sum + i i = i + 2 Loop Label3.Text = sum	Dim N, i, sum As Integer N = TextBox1.Text For i = 1 To N Step 2 sum = sum + i Next Label3.Text = sum



تعريف الإجراء : هو مجموعة من الأوامر والتعليمات (Code) له اسم معين عند استدعاء هذا الاسم يتم تنفيذ الأوامر والتعليمات .

أنواع الإجــراءات فى لغة Visual Basic.net :-

١- الإجراء SUP لا يعود بقيمة .
٢- الإجراء Function يعود بقيمة .

أولاً: الإعلان عن الإجراء SUP:-

عبارة عن وجود كود معين سيتكرر كتابته فى أكثر من مكان داخل التصنيف Class مثال كود يتكرر فى الاداة Button 1 و الاداة Button 2 لعرض الاعداد الفردية والزوجية نفس الكود يتكرر فى الإجرائين مع تغير قيمة البداية فى الاعداد الفردية البداية تكون ١ وفى الاعداد الزوجية البداية تكون ٢ وتستخدم Sup لكتابته مرة واحدة لسهولة قراءة الكود والإجراء يكتب داخل Form بالكود يمكن استخدام اى عدد من الوسائط
يستدعى الإجراء بكتابة اسمه ووضع قوسين فى نهاية الاسم
يتم الاعلان عن الإجراء مرة واحدة ويمكن أستدعاءه اى عدد من المرات

الصيغة العامة للإعلان عن Sup :

Sup Name (Parameters) أسم الاجراء القيم التى تستقبل عند استدعاء الاجراء والتى تستخدم فى الكود
Code مجموعة الأوامر التى ستنفذ عند استدعاء الإجراء
End Sup

١- Name – ١ : تعبر عن اسم الاجراء **Procedure**
٢- Parameters : عبارة عن القيم التى يستخدمها داخل الكود الاجراء عند استدعاء الاجراء
٣- Code : مجموعة الأوامر والتعليمات التى ستنفذ عند استدعاء الاجراء Sub

```
Private Sub Button1_Click(ByVal sender As Sys
    Dim X As Integer
    Label1.Text = ""
    For X = 1 To 20 Step 2
        Label1.Text = Label1.Text & " " & X
    Next
End Sub
```

```
Private Sub Button2_Click(ByVal sender As Sys
    Dim X As Integer
    Label1.Text = ""
    For X = 2 To 20 Step 2
        Label1.Text = Label1.Text & " " & X
    Next
```

Form1
2 4 6 8 10 12 14 16 18 20
الاعداد الزوجية
الاعداد الفردية

لكن بالكود السابق نلاحظ كتابة الكود مرتين باختلاف نقطة البداية لعرض الاعادد الفردية والزوجية لذلك سوف نستخدم الاجراء لكتابة الكود مرة واحدة واستدعائه اكثر من مره من خلال
الاعلان عن الاجراء Sub واستدعاءه لتجنب تكرار كتابة كود اكثر من مره

```
Public Class Form1
    Dim total As Integer

    Sub Amalmostafa(ByVal a As Single, ByVal b As Single, ByVal c As Single)
        Dim i As Integer
        Label1.Text = " "
        For i = a To b Step c
            Label1.Text = Label1.Text & " " & i
        Next
    End Sub


    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System
        Amalmostafa(1, 10, 2)
    End Sub

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System
        Amalmostafa(2, 10, 2)
    End Sub
End Class
```

تم الاعلان عن إجراء باسم **amalmostafa** يحتوى على الكود المكرر ويتم تنفيذ الاجراء بكتابة اسمه داخل التصنيف مثل كتابة اسمه داخل **Button1** و **Button2**

ولابد من الاعلان عن **Parameters**

نلاحظ عند كتابة الكود بهذه الطريقة يتم تكرار الاعداد الفردية فقط دون الزوجية لان نقطة البداية ١ ولتجنب ذلك سوف نعلن عن وسائط

```
Sub Amalmostafa()
    Dim i As Integer
    Label1.Text = " "
    For i = 1 To 10 Step 2
        Label1.Text = Label1.Text & " " & i
    Next
End Sub
```

لذلك نعلن عن الاجراء باسم **amalmostafa** ثم الوسائط ويعبر **a** عن البداية و **b** عن نهاية و**c** عن مقدار الزيادة

Sub Amalmostafa(ByVal a As Single, ByVal b As Single, ByVal c As Single)

وفى كل مره يتم استدعاء الاجراء يتم وضع قيمة البداية بلا من **a** ووضع النهاية بدلا من **b** ووضع مقدار الزيادة بدلا من **c** كالتالى **Amalmostafa(1, 10, 2)** لعرض الاعداد الفردية و **Amalmostafa(2, 10, 2)** لعرض الاعداد الزوجية وهكذا لعرض اى اعداد مث عرض الاعداد من ١٥ الى ٦٠ بمقدار الزيادة ٥ **Amalmostafa(15, 60, 5)**





الدالـــــة Function :- عبارة عن مجموعة من الأوامر تحت اسم معين يفضل ان يكون معبرا عن وظيفتها يتم تطبيقها على وسائط وتعود بقيمة

الصيغة العامة للاعلان عند دالة الدالة function :

١- تبدأ الدالة بكلمة function
٢- Nem تشير إلى أسم الدالة الذى سوف تستدعى من خلاله
٣- Parameters هى الوسائط التى تستخدم فى الكود
٤- Data Type تحد نوع البيان القيمة الراجعة من الدالة
٥- Code هى الأوامر والتعليمات التى ستنفذ عند استدعاء الدالة
٦- ترجع قيمة باستخدام Return
٧- Value تشير الى القيمة الراجعة من الدالة
٨- End Function هى نهاية الدالة

الصيغة العامة للإعلان عن الدالة function:

```
Function Name ( Parameters ) As Data Type
Code   مجموعة الأوامر والتعليمات المطلوب تنفيذها
Return Value
End function
```

لاحــــظ أن :

- الدوال Function لا يمكن تخصيص قيمة لها
- الدوال Function يتم استدعاءها فتنتج قيمة تخزن بها ومن ثم يمكن استخدام هذه القيمة
- الدوال Function يفضل تسميتها تسمية تدل على وظيفتها

- **المتغيرات (Variables):** يمكن تخصــــيص قيم لها <u>أثناء الإعلان وأثناء ســــير تنفيذ تعليمات البرنامج</u>، وكذلك استخدام القيم المخزنة بها.
- **الثوابت (Constants):** يمكن تخصيص قيم لها أثناء الإعلان <u>**فقط**</u> وكذلك استخدام القيم المخزنة بها.
- **الدوال (Function):** تستدعى فتعود بقيمة في ضوء القيم المخصصة لها.

مثال : أعداد مشروع لحساب مجموع عددين واظهار الناتج :

```
Public Class Form1
    Function Sum(ByVal first As Single, ByVal second As Single) As Single
        Dim total As Single
        total = first + second
        Return total
    End Function

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As Sys
        Dim X As Single = TextBox1.Text
        Dim Y As Single = TextBox2.Text
        Label4.Text = Sum(X, Y)
    End Sub
End Class
```

Form1
العدد الاول 85
+
العدد الثانى 43
النتيجة 128

شرح الكود : بداية الاعلان عن دالة Function:
Sum اسم الدالة
First As Single, second As Single تم الاعلان عن قينتين من النوع العشرى
نوع الدالة رقم عشرى Single
الإعلان عن متغير Total من النوع الرقمى العشرى
تخصيص ناتج جمع القيمتين (first ، second) للمتغير Total
إرجاع قيمة المتغير Total باستخدام الأمر Return
نهاية الدالة End Function
تم تخزين قيمة المتغير Total فى اسم الدالة Sum



الانترنت له جانبين :

١- جانب ايجابى مثل نتعلم – نتثقف – نتواصل
٢- جانب سلبى : الحصول على معلومات خطا – انتهاك الخصوصية – انتحال الشخصية – سرقة حسابك البريدى – تعرض جهازك لمخاطر كالاصابة بفيروسات او برامج تجسس او برامج القرصنة

تعريف التعدى الإلكترونى :-

عبارة عن سلوك عدوانى متعمد من شخصباستخدام الوسائط الإلكترونية بغرض (التحرش ـ المضايقة ـ الأحراج ـ التخويف ـ التهديد ـ الخ)

ثانياً: الوسائط الإلكترونية للتعدى:-

١- البريد الكترونى E – mail . امكانية ارسال واستقبال الرسائل الالكترونية والرد عليها وارفاق ملفات بتلك الرسائل
٢- المنتديات الإلكترونية Forms . تستخدم لطرح موضوعات للمناقشة فيتم ارسال المشاركة وتجد ردود من الآخرين
٣- الرسائل الفورية instant message . هى اتصال فورى بين طرفين أو اكثر بالكتابة أو المحادثة الصوتية أو المرئية
٤- المدونات الإلكترونى Blooding . هو سجل خاص الكترونى يدون به صاحب المدونة ما يشاء او مجموعة من صفحات الويب تحتوى على نصوص وصور ورسومات و اصوات وفيديو
٥- المواقع التواصل الاجتماعى مثل Face Book . تسمح بنشر معلومات خاصة للشخص صاحب الحساب

ثالثاً: أشكال التعدى الإلكترونى :-

١- التخفى الإلكترونى (Anonymity) : عبارة عن استخدام أسماء مستعارة تخفى شخصية المتعدى الإلكترونى بغرض الإفلات من العقاب .
٢- المضايقات الإلكترونية (Harassmennnnt) : عبارة عن رسائل عدائية موجهة ضد شخص أو أكثر .
٣- الملاحقة الإلكترونية (Cyber stalking) : هى شكل من أشكال المضايقات الالكترونية لكن بشكل متكرر.
٤- السب أو القذف الإلكترونى (Flaming) : عبارة عن نشر كلمات عدائية ومبتذلة ضد شخص أو أكثر .
٥- التشهير الإلكترونى (Outing) : عبارة عن نشرمعلومات عن شخص محدد أو أكثر بشكل مسيئ .
٦- الاستثناء الإلكترونى (Exclusion) : عبارة عن تجاهل شخص أو أكثر من خلال وسائط الكترونية .
٧- التهديد الإلكترونى (Cyber threats) : عبارة عن إرسال رسائل إلكترونية تحمل تهديد أو وعيد لشخص أو أكثر .

رابعاً: كيف تحمى نفسك من التعدى الإلكترونى ؟

١- لا تشارك أحد كلمة المرور .
٢- إعداد كلمة مرور لا تستنتج .
٣- عدم نشر أى بيانات خاصة .
٤- عدم حذف رسائل التعدى .
٥- عدم مقابلة أحد تعرفت عليه من خلال الانترنت .
٦- عدم إرسال رسائل وأنت فى حالة غضب .
٧- اطلاع ولى الأمر بما يضايقك عند استخدام الانترنت .
٨- إنزال البرامج من الانترنت يكون تحت إشراف معلمك أو ولى الأمر .